



Borcelle
PIZZA RESTAURANT

Special Delicious

PIZZA

Life is short to enjoy pizza !



Hello!.

My name is khushbu chudhari ,and i'am excited to present my analysis of pizza sales using MYSQL With passion for data and analytics,I have applied MYSQL to understand key, trends ,insights , and strategies that can help optimize pizza sales.In this presentation .i will you through data -driven decisions that can drive growth and customer satisfaction.



Insights

Retrieve the total number of orders placed.

Calculate the total revenue generated from pizza sales.

Identify the highest-priced pizza.

Identify the most common pizza size ordered.

List the top 5 most ordered pizza types along with their quantities.

Join the necessary tables to find the total quantity of each pizza category ordered.

Determine the distribution of orders by hour of the day.

Join relevant tables to find the category-wise distribution of pizzas.

Group the orders by date and calculate the average number of pizzas ordered per day.

Determine the top 3 most ordered pizza types based on revenue.

Calculate the percentage contribution of each pizza type to total revenue.

Analyze the cumulative revenue generated over time.

Determine the top 3 most ordered pizza types based on revenue for each pizza category

Q-1 Retrieve the total number of orders placed.

```
SELECT  
    COUNT(ORDER_ID) AS TOTAL_ORDERS  
FROM  
    ORDERS;
```

TOTAL_ORDERS

21350

Q-2 Calculate the total revenue generated from pizza sales.

```
SELECT  
    ROUND(SUM(orders_details.quantity * pizzas.price),  
          2) AS total_sale  
FROM  
    orders_details  
        JOIN  
    pizzas ON pizzas.pizza_id = orders_details.pizza_id;
```

total_sale

817860.05

Q-3 Identify the highest-priced pizza.

```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

	name	price
→	The Greek Pizza	35.95

Q-4 Identify the most common pizza size ordered.

```
pizzas.size,  
COUNT(orders_details.order_details_id) AS order_count  
FROM  
pizzas  
JOIN  
orders_details ON pizzas.pizza_id = orders_details.pizza_id  
GROUP BY pizzas.size  
ORDER BY order count DESC;
```

	size	order_count
	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

Q-5 List the top 5 most ordered pizza types along with their quantities.

```
SELECT pizza_types.name , sum(orders_details.quantity) as quantity
FROM pizza_types JOIN pizzas
ON pizza_types.pizza_type_id=pizzas.pizza_type_id
JOIN orders_details
ON orders_details.pizza_id=pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity desc
limit 5;
```

name	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Q-6 Join the necessary tables to find the total quantity of each pizza category ordered.

```
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

category	quantity
Classic	1488
Supreme	11987
Veggie	11649
Chicken	11050

Q-7 Determine the distribution of orders by hour of the day

```
SELECT  
    HOUR(order_time) AS HOUR, COUNT(order_id) AS order_count  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

HOUR	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

Q-8 Join relevant tables to find the category-wise distribution of pizzas.

```
SELECT  
    category, COUNT(name)  
FROM  
    pizza_types  
GROUP BY category;
```

category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

Q-9 Group the orders by date and calculate the average number of pizzas ordered per day.

```
SELECT
    round(avg(quantity)) AS avg_pizzas_per_day
FROM
    (SELECT
        orders.order_date, SUM(orders_details.quantity) AS quantity
     FROM
        orders
     JOIN orders_details ON orders.order_id = orders_details.order_id
     GROUP BY orders.order_date) AS order_quantity;
```

avg_pizzas_per_day
138

Q-10 Determine the top 3 most ordered pizza types based on revenue.

```
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
    orders_details
        JOIN
    pizzas ON orders_details.pizza_id = pizzas.pizza_id
        JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

Q-11 Calculate the percentage contribution of each pizza type to total revenue.

```
SELECT
    pizza_types.category,
    ROUND(SUM(orders_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(orders_details.quantity * pizzas.price),
        2) AS total_sale
    )
    FROM
        orders_details
        JOIN
            pizzas ON pizzas.pizza_id = orders_details.pizza_id) * 100,
    2) AS revenue
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
        orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

Q-12 Analyze the cumulative revenue generated over time.

```
SELECT order_date,
       sum(revenue) over(ORDER BY order_date) AS cumulative_revenue
  FROM
    (select orders.order_date,
           sum(orders_details.quantity * pizzas.price) AS revenue
      FROM orders_details
     JOIN pizzas
       ON orders_details.pizza_id=pizzas.pizza_id
     JOIN orders
       ON orders.order_id=orders_details.order_id
   GROUP BY orders.order_date) as sales;
```

order_date	cumulative_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.35000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.30000000003
2015-01-14	32358.70000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.60000000006
2015-01-19	43365.75000000001
2015-01-20	45763.65000000001
2015-01-21	47804.20000000001
2015-01-22	50300.90000000001
2015-01-23	52724.60000000006
2015-01-24	55013.85000000006
2015-01-25	56631.40000000001

Q-13 Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```
SELECT name,revenue
from
(SELECT category,name,revenue,
rank() over(partition by category order by revenue DESC) AS RN
FROM
(SELECT pizza_types.category,pizza_types.name,
sum((orders_details.quantity )* pizzas.price) as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id=pizzas.pizza_type_id
join orders_details
on orders_details.pizza_id=pizzas.pizza_id
GROUP BY pizza_types.category,pizza_types.name) AS A) AS B
where rn <= 3;
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5